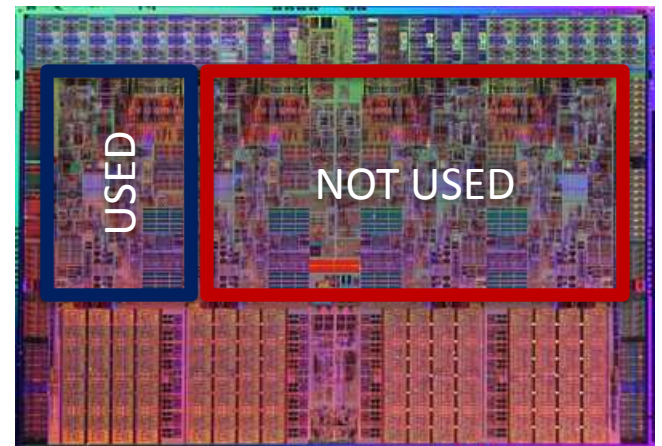


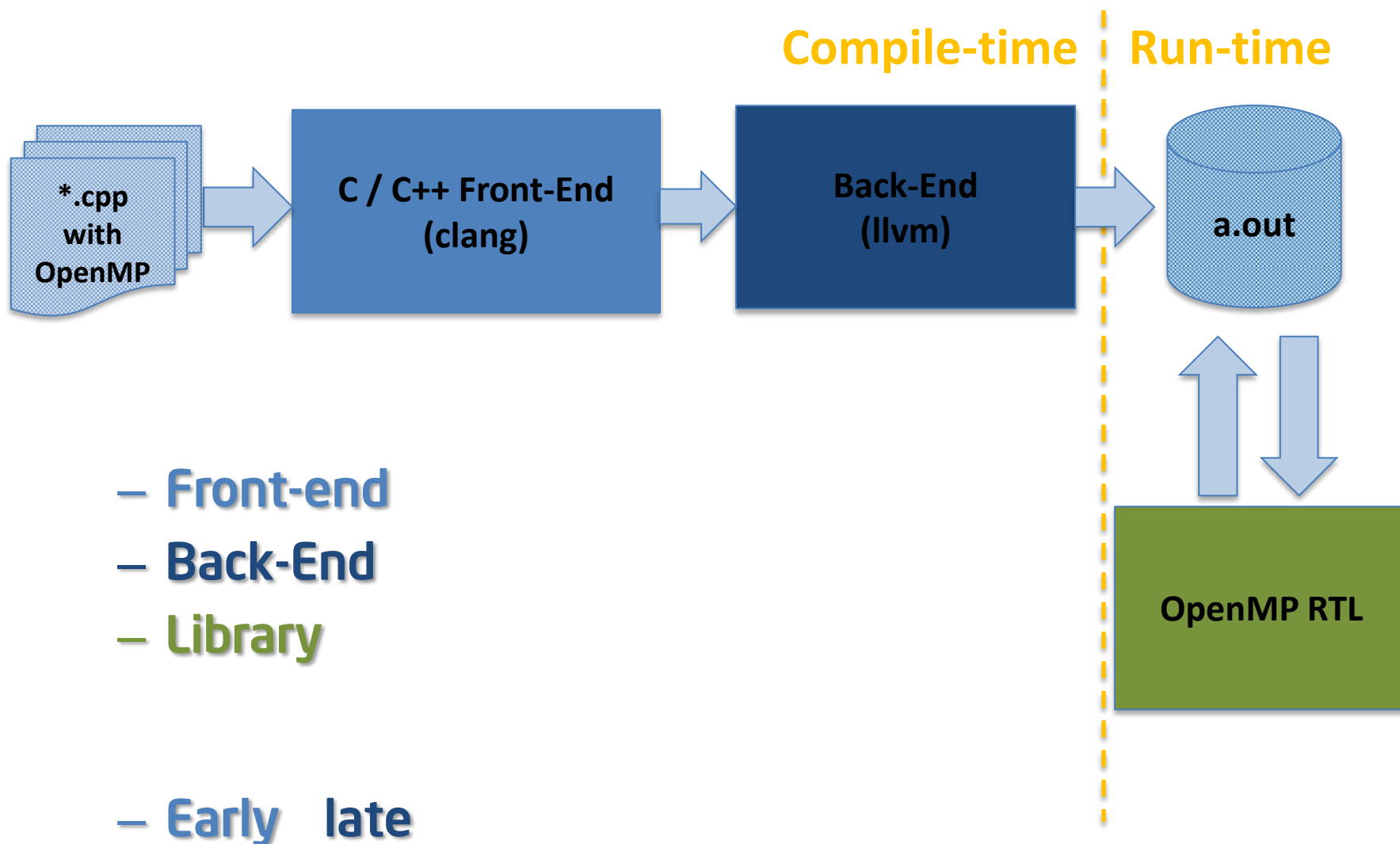
- Industry-wide standard

```
#pragma omp parallel for  
for (i = 0; i < N; i++)  
{  
    ...  
}
```

- www.openmp.org

- **Anyone**

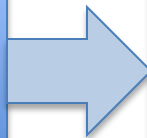




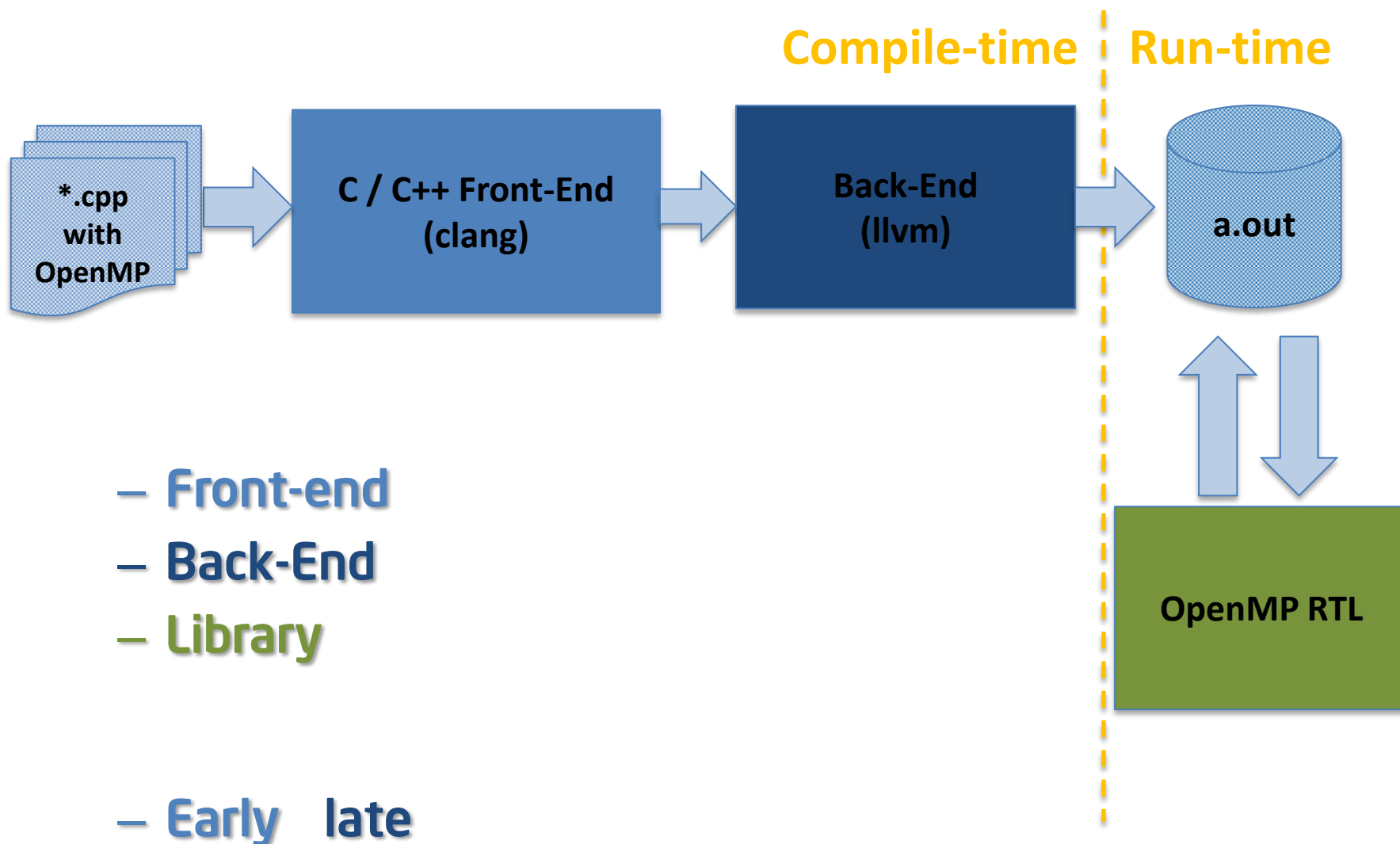
Early Late

front-end back-end

```
float a,x,y,z;  
#pragma omp parallel for  
for (i = 0; i < N; i++) {  
    a[i] = x * y * z;  
    ... // rest of loop  
}
```



```
omp_parallel_for(0, N,  
N/omp_get_num_threads(), forb)  
...  
void forb(int L, int U, R *r) {  
    for (i = L; i < U; i++) {  
        r->a[i] = r->x * r->y * r->z;  
        ... // rest of loop  
    }  
}
```

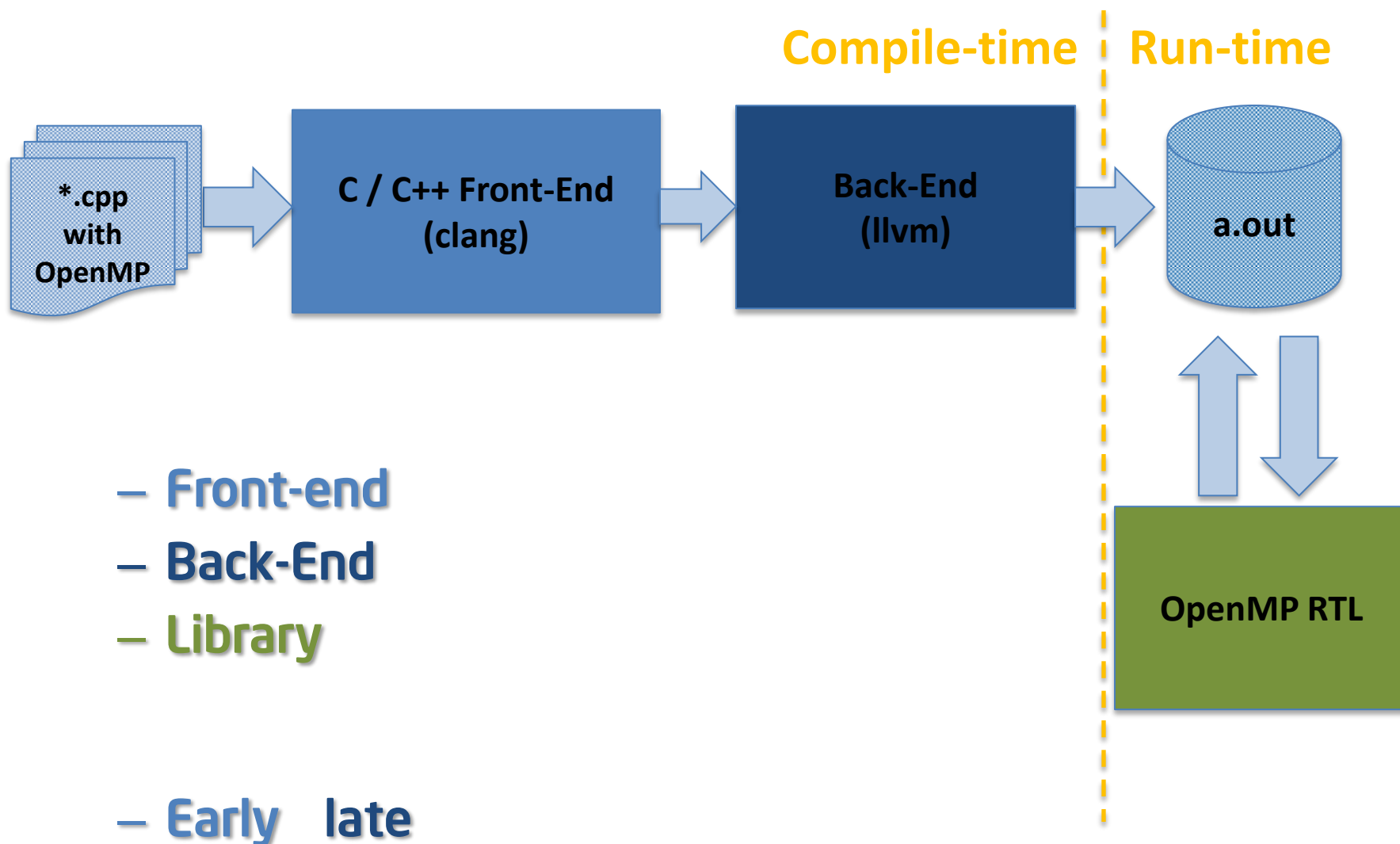



GPLv3*

- Permissively licensed

BSD license

www.openmpRTL.org



C++11 attributes

standard pragma

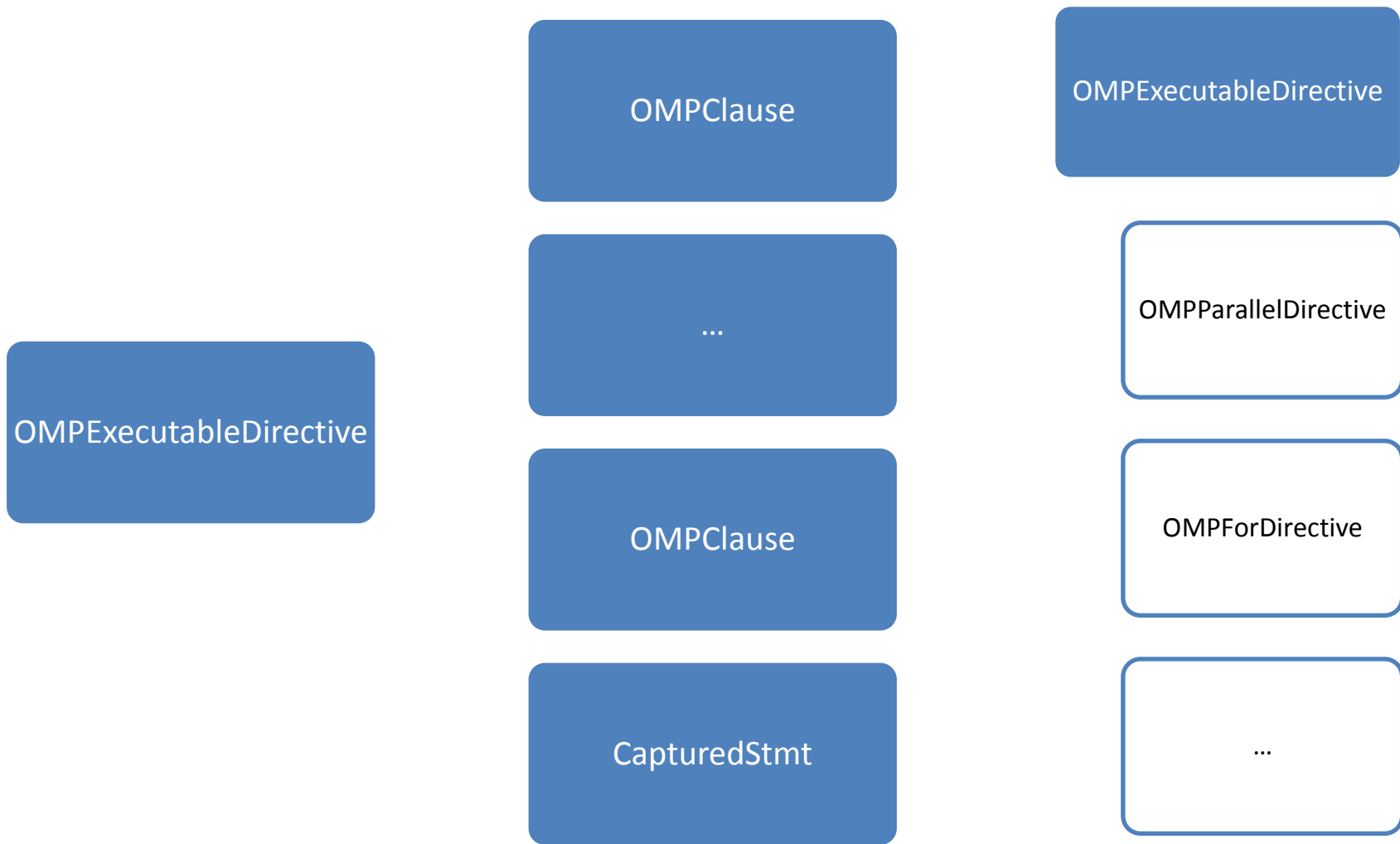
parsing harness

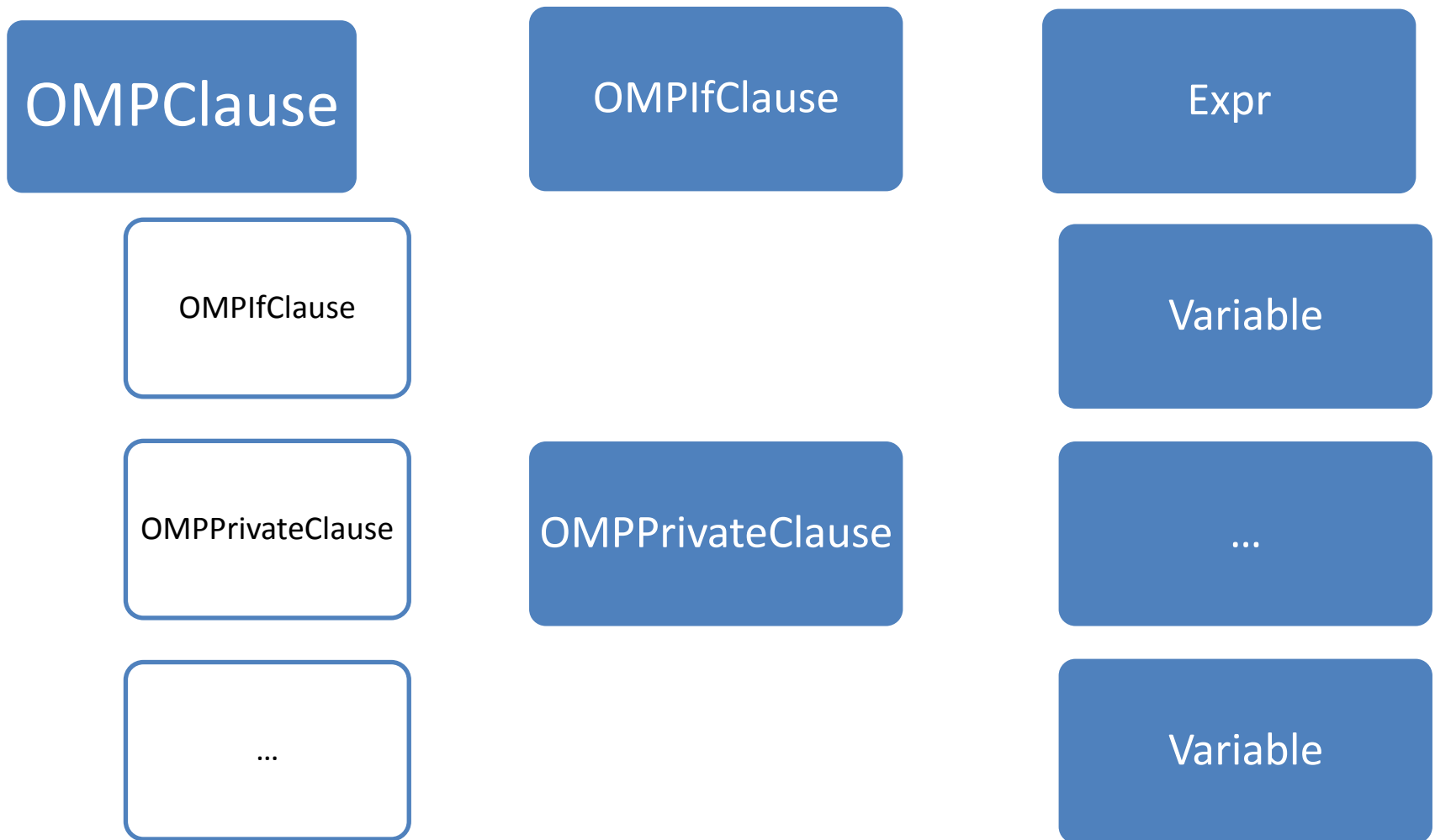
OMPThreadPrivateDecl

Variable

...

Variable





- **Statements Structured Statements**

CapturedStmt

- **Private constructed**
- **Shared captured**

```
#pragma omp parallel if(a) private(argc,b)
foo();
```



```
-OMPParallelDirective <line:9:2, col:43>
| |-OMPIfClause <col:22, col:27>
| | `ImplicitCastExpr <col:25> '_Bool' <IntegralToBoolean>
| | `ImplicitCastExpr <col:25> 'int' <LValueToRValue>
| | `DeclRefExpr <col:25> 'int' lvalue Var 'a' 'int'
| |-OMPPrivateClause <col:28, col:43>
| | |-DeclRefExpr <col:36> 'int' lvalue ParmVar 'argc' 'int'
| | `DeclRefExpr <col:41> 'int' lvalue Var 'b' 'int'
| `CapturedStmt <line:10:2, col:7>
| `CallExpr <col:2, col:7> 'void'
| `ImplicitCastExpr <col:2> 'void (*)(void)' <FunctionToPointerDecay>
| `DeclRefExpr <col:2> 'void (void)' lvalue Function 'foo' 'void (void)'
```

generated record

auto-

functions

captured

variables

- Thank you

welcomed

INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS”. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Copyright © , Intel Corporation. All rights reserved. Intel, the Intel logo, Xeon, Xeon Phi, Core, VTune, and Cilk are trademarks of Intel Corporation in the U.S. and other countries.

Optimization Notice

Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

